

Mobile data gathering with hop-constrained clustering in underwater sensor networks

Ghoreyshi, Seyed Mohammad; Shahrabi, Alireza; Boutaleb, Tuleen; Khalily, Mohsen

Published in:
IEEE Access

DOI:
[10.1109/ACCESS.2019.2897872](https://doi.org/10.1109/ACCESS.2019.2897872)

Publication date:
2019

Document Version
Publisher's PDF, also known as Version of record

[Link to publication in ResearchOnline](#)

Citation for published version (Harvard):
Ghoreyshi, SM, Shahrabi, A, Boutaleb, T & Khalily, M 2019, 'Mobile data gathering with hop-constrained clustering in underwater sensor networks', *IEEE Access*, vol. 7, pp. 21118-21132.
<https://doi.org/10.1109/ACCESS.2019.2897872>

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

If you believe that this document breaches copyright please view our takedown policy at <https://edshare.gcu.ac.uk/id/eprint/5179> for details of how to contact us.

Mobile Data Gathering with Hop-Constrained Clustering in Underwater Sensor Networks

SEYED MOHAMMAD GHOREYSHI¹, (Member, IEEE), ALIREZA SHAHRABI¹, (MEMBER, IEEE), TULEEN BOUTALEB¹, (MEMBER, IEEE), AND MOHSEN KHALILY², (Senior member, IEEE)

¹School of Computing, Engineering and Built Environment, Glasgow Caledonian University, Glasgow, UK (e-mail: {Seyed.MohammadGhoreyshi, A.Shahrabi, T.Boutaleb}@gcu.ac.uk)

²Institute for Communication Systems, Home of the 5G Innovation Centre, University of Surrey, Guildford GU2 7XH, U.K. (m.khalily@surrey.ac.uk)

Corresponding author: Mohsen Khalily (e-mail: m.khalily@surrey.ac.uk).

ABSTRACT Recent years have witnessed growing attention to the mobile data gathering schemes using Autonomous Underwater Vehicle (AUV) in Underwater Sensor Networks (UWSNs). In this paper, a novel Cluster-based Mobile Data Gathering scheme (CMDG) for large-scale UWSNs is presented to make a trade-off between the data gathering latency and energy saving. To cluster the acoustic sensors and cover their heads with the shortest possible tour, we first formulate it into an optimization problem, and then propose two efficient algorithms to obtain the near-optimal solutions in the less computational time. We also discuss the impact of topology change on the CMDG performance. To the best of our knowledge, CMDG is the first AUV tour planning scheme which is able to deal with the sensors mobility in UWSNs. CMDG is scalable and also applicable in both connected and disconnected networks. In terms of energy-latency trade-off, CMDG can effectively keep the tour length short while prolonging the network lifetime compared to those of existing mobile data gathering schemes. The effectiveness of CMDG is validated through an extensive simulation study which reveals the performance improvement in the energy saving, data gathering latency and packet delivery ratio.

INDEX TERMS Underwater sensor networks, Autonomous underwater vehicle, Clustering

I. INTRODUCTION

Nowadays, underwater acoustic sensor networks have gained a great interest in a wide range of aquatic applications, such as environmental monitoring, pollution monitoring, disaster prevention, battlefield surveillance, exploration of ocean resource, etc [1]–[3]. Underwater sensors are spatially distributed in a marine environment to gather data and deliver them to a destination, which may be a static sink, or a mobile sink (e.g. AUV) [4]. Surface sink then delivers the accumulated information to the monitoring centre via the terrestrial radio links for further analysis, as shown in Fig. 1. Due to the quick absorption of radio waves in water, underwater sensors take the advantage of acoustic transmission. However, a data packet should be transmitted with the higher power to cope with the path loss and high bit error rate which makes the energy consumption as a major concern [5]–[7].

In a multi-hop data forwarding model, the energy of sensors near to the static sink is depleted quickly because

of relaying a high number of data packets generated from different sources. Therefore, these sensors may die earlier than others and consequently affecting the network connectivity. The connectivity problem can be exacerbated in a large-scale UWSN [8], [9]. Nonetheless, data gathering using AUV is more appropriate for large-scale networks due to decreasing the number of transmissions and balancing the energy consumption [10].

AUV is a mobile sink equipped with a powerful transceiver, moving through the underwater field to continuously gather data packets from sensors [11], [12]. The data collection tour is periodically started from a static base station, followed by gathering data packets from sensors, and completed by delivering data packets to the static base station [13]. Using an AUV contributes to prolonging the lifetime of acoustic sensors since any packet relay is bounded within a given number of hops. It also equally distributes energy consumption in order to preserve the network coverage [14].

An optimal AUV tour planning should always make a trade-off between data gathering latency and energy saving [15], [16]. The maximum energy saving for sensors can be achieved when the AUV visits each sensor directly to collect the data. However, visiting all sensors in the field requires a longer tour which can increase the data gathering latency. Hence, the local aggregation in a subset of sensors, as cluster heads, can reduce the data gathering latency. The energy consumption and packet loss are reduced by bounding the number of transmission hops from members to CHs [17], [18].

In recent years, a variety of mobile data gathering schemes have been proposed for the terrestrial wireless sensor networks [17], [19]–[22]. However, due to the different characteristics of UWSNs, such as three-dimensionality, high propagation delay, limited bandwidth, and path loss, mobile data gathering schemes proposed for terrestrial networks are impractical in the underwater environment [23], [24]. Taking into account the different characteristics of the underwater environment, designing efficient mobile data gathering schemes for UWSNs is crucial.

In this paper, a new Cluster-based Mobile Data gathering scheme (CMDG) is proposed to improve the packet delivery ratio, energy consumption, latency, and tour length compared to those of other proposed protocols. The CMDG selects a subset of acoustic sensors as Cluster Heads (CHs) to collect data from affiliated members. The packet relay is bounded within a few numbers of hops to reduce the energy consumption and the chance of collisions at sensors. The aggregated data is then collected using an AUV visiting them in a certain order. To this end, we first formulate the problem into an optimization problem, and then propose a centralised and distributed clustering algorithm to obtain the near-optimal solutions. In centralised clustering, we take advantage of randomness and greediness to create some local solutions, and to find an acceptable solution among them using less computational time. In distributed clustering, underwater sensors compete to be a cluster head based on their priority in a distributed manner.

We also show that CMDG can keep its performance high in the presence of sensor mobility while many other mobile data gathering approaches are impractical to be used in such a dynamic environment. CMDG supports the proactive maintenance of routes within each cluster, and route discovery is only required when a sensor departs from its original cluster. In CMDG, sensor mobility is handled locally which leads to perform properly over a variety of conditions. Using this technique, the clustering phase does not need to be repeated frequently. Furthermore, it is shown that the mobility information can be used to improve the efficiency of AUV tour planning and maintenance.

This paper is an extended version of our previous work [25], [26]. The present paper extends the previous work with formulating the problem into an optimisation problem, discussing the problem complexity, and proposing a mobility handling mechanism. We also propose additional results

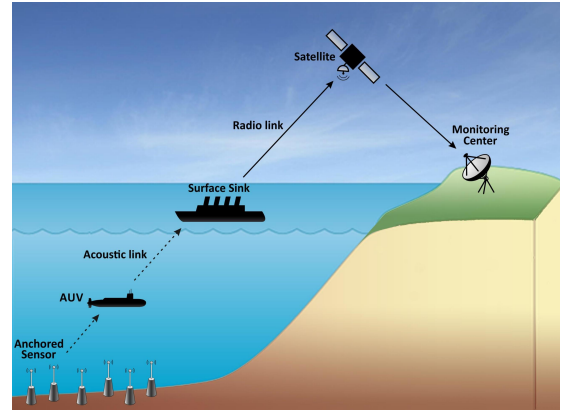


FIGURE 1: UWSN Architecture

related to the network parameters impacts such as sensor mobility, relay hop bound, and different network sizes. The mobile data gathering scheme is also discussed with more details and examples. The rest of this paper is organized as follows. In Section II, we motivate this work and present the related work. In Section III, we provide a detailed description of the system model. In Section IV, CMDG is presented in detail. In Section V, the impact of sensor mobility on CMDG performance is discussed. Section VI presents and discusses the results of our simulation study. In Section VII, we conclude the paper.

II. MOTIVATION AND RELATED WORK

In this section, we motivate this work and present the related work.

A. MOTIVATION

In UWSNs, there are two main categories of data forwarding: pressure based and geographic greedy data forwarding [4]. The pressure based data forwarding can only be used in deep scenarios where the data packets can be forwarded towards the surface using the depth information [27], [28]. However, in a flat network in which sensors are distributed in the same depth, but different horizontal coordinates, pressure based approaches are impractical.

In the category of geographic greedy forwarding, each sensor relays the data packet closer to the destination in each hop using the geographic position information [29]. An example of a geographic greedy forwarding is shown in Fig. 2. However, there are some constraints which can limit the performance of a geographic greedy forwarding in the underwater environment. First of all, GPS does not work in an underwater environment. Therefore, there is a need to use more complex localisation techniques to obtain the geographic coordinates for each sensor. Second, some packets must be relayed in a high number of hops to reach the destination. As can be seen in Fig. 2, some paths between a source and the destination include so many hops and the packet failure probability increases in these kinds of paths

because UWSN is a noisy and lossy environment. Moreover, the energy and the network resources are wasted because there is no bound on the number of packet transmissions. Even worse, some sensors which are closer to the destination become a bottleneck in terms of energy because they must relay lots of packets from different sources to the destination.

Finally, the greedy forwarding approach may fail because of communication voids [4], [30]. In this case, a data packet may be dropped because a forwarding sensor cannot find any other neighbouring sensor closer to the destination. As an example, this phenomenon is presented in Fig. 2 where sensor v is a void sensor because it cannot find any neighbouring sensor closer to the destination. Data packets are therefore stuck in the void node and cannot proceed. The void problem in geographic greedy forwarding decreases the network performance in terms of the packet delivery ratio and energy.

Mobile data gathering, instead, is quite useful in these kinds of scenarios to cope with the void, energy waste, and high bit error rate problems. A number of studies have been reported in the literature to use mobile data gathering mechanisms to improve energy efficiency in terrestrial networks [17], [19]–[21]. However, these mechanisms cannot be used directly in an underwater environment without considering their challenging constraints. Over the past few years, some mobile data gathering techniques are specifically proposed for UWSNs. In what follows, we briefly review those underwater mobile data gathering protocols that are based on some realistic assumptions.

B. RELATED WORK

Mobicast [31] is a mobile data gathering scheme in which the AUV traverses a predetermined circular path to gather data packets from sensors in different geographic areas called 3-D zone of references (3-D ZOR). The data packets are directly forwarded to an AUV using multi-hop transmissions and without using any clustering mechanism. While the AUV investigates sensors within a 3-D ZOR_{*t*}, it should notify the sensors within 3-D ZOR_{*t+1*} to enter the active mode to be ready for the arrival of AUV. However, some limitations can confine the data gathering performance. For instance, to deal with the sensor mobility and finding the alternative paths, a larger coverage area is also considered to surround the 3-D ZOR. This coverage area may involve more relaying sensors and hence more energy consumption. Moreover, data collection from all sensors in the sensing field is not possible because only sensors within 3-D ZORs are investigated.

In AEERP (AUV aided Energy Efficient Routing Protocol) [32], an AUV is employed to traverse a predetermined elliptical trajectory in each data gathering round. A group of sensors is selected as the gateway to receive and deliver the data packets to an AUV. Each gateway sensor is selected based on the distance to the AUV trajectory and also the residual energy. Each member is then allocated to a gateway using a Shortest Path Tree (SPT). Nevertheless, there is no bound on the hop distance from members to a gateway, which causes



FIGURE 2: Geographic greedy data forwarding

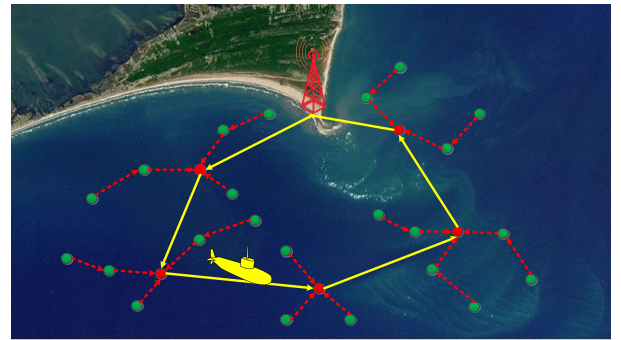


FIGURE 3: Mobile data gathering within two hops

higher energy consumption. In terms of the void problem, a group of nodes may be isolated and cannot reach any gateway near to the AUV trajectory. The packets generated by these nodes are dropped as no recovery mechanism is considered.

In AURP (AUV-aided Underwater Routing Protocol) [33], multiple AUVs are employed to gather data from predetermined gateway nodes. The gateway nodes should periodically broadcast their interest in receiving data to be used by sensors for routing paths establishment. AURP inherently addresses the void problem as the reachability information to the sink and gateways is distributed among the nodes. However, AURP is only suitable for a static network as it cannot deal with the node mobility. The procedure of gateway selection with its overhead has not been investigated thoroughly in AURP. Furthermore, an AUV trajectory is a fixed elliptical path which decreases the flexibility when facing different network densities and topologies.

AUV_PN [34] is another mobile data gathering scheme in which the network is divided into a number of clusters. A cluster is further divided into several sub-clusters with a Path Node (PN) to gather data from its members. During the data gathering operation, the AUV travels to the nearest CH to retrieve the list of PNs and then visits each PN to gather data. After visiting all PNs in a cluster, the AUV travels to the next nearest CH and repeat the same procedure until all clusters are visited. Finally, the AUV returns to the base station to deliver all aggregated data. However, the tour constructed by the AUV crosses over itself and is not optimal. Having a

larger network, the data packets should be transmitted with higher power because of the high distances between PNs and members. The procedure of network-partitioning is also complicated and consumes a lot of energy.

III. SYSTEM MODEL

In this section, the network architecture and physical layer model are described in detail. A typical mobile data gathering UWSN has been shown in Fig. 1.

A. NETWORK ARCHITECTURE

Acoustic sensors are distributed in a 3D network topology. The depth of sensors is fixed and it is initially assumed that sensors are static or anchored to the subsea-floor [31], [34]. Later, a realistic sensor mobility model is considered to evaluate the network performance under continuous topology changes. Sensors are homogeneous in terms of power and transmission range. The Received Signal Strength Indicator (RSSI) can be used by sensors to measure the pairwise distances [35]. Sensors are also unaware of their full geographical coordinates.

A mobile sink (AUV) is deployed at a fixed depth above the acoustic sensors with the ability to move in all directions. The AUV can obtain the full geographical coordinates of each sensor by marking the location where it receives data from [36]. The gathered data by the AUV should be delivered to a static sink on the water surface. These assumptions have been widely used in the literature [31], [34]–[36].

B. PHYSICAL LAYER MODEL

The BELLHOP ray tracing model [37], [38] can represent the underwater acoustic channel model by performing acoustic ray tracing for a given speed of sound profile and absorbing boundary conditions. In this model, the acoustic fields can be computed via Gaussian beam tracing to predict the ray trajectories of arrivals using parameters such as sound speed profile, signal frequency, the depth of source and receiver, the horizontal distance between sensors, and bathymetric information [39]. The behaviour of this propagation model is very similar to experimental studies for acoustic propagation in underwater environments [40].

To implement the BELLHOP model, the parameters are adopted from [41] to carry out ray tracing and transmission loss. The channel consists of an iso-speed sound velocity profile with the speed of sound in water equal to 1500 m/s, and bottom sound speed equal to 1800 m/s. The seawater density and sea-bottom density are taken as 1024 kg/m³ and 1843 kg/m³, respectively.

IV. CMDG DETAILS

In this section, our proposed scheme (CMDG) is presented.

A. OVERVIEW

The tour-planing schemes have gained much attention in UWSNs by finding an efficient way of regularly collecting data from sensors. The CMDG is a cluster-based scheme in

which a subset of sensors are selected as Cluster Heads (CHs) to receive data from the affiliated sensors within a limited number of hops and deliver the aggregated data to an AUV when it arrives within a single-hop vicinity of the CH. The CMDG finds a short tour for an AUV to visit each CH in a certain order. An example of a mobile data gathering scheme with a constraint of two hops distance from members to a CH is depicted in Fig. 3.

The relay hop count between members and CHs should be bounded for a number of reasons. First, energy efficiency can be obtained by confining the number of packet transmissions from members to CHs [17]. Second, the chance of packet failure is increased by forwarding a packet over several hops in a noisy channel of the underwater environment [42]. Third, the buffering capacity of each sensor is limited, and consequently, a high number of sensors cannot be allocated to one CH.

The relay hop bound, d , is a system parameter which can be set based on the application priorities on the latency and energy saving. The d can be set to a small value for delay-insensitive applications to save more energy at sensors. Mobile data gathering schemes are especially appropriate for applications which can tolerate the delay [17].

For the clustering and tour planning, the problem is first formulated into an optimisation problem, and then two efficient algorithms are proposed to solve the problem with less computational time. In the centralised approach, CHs are selected using a greedy iterative search method while in the distributed approach, sensors compete with each other to become a CH.

Although the data gathering phase consumes a significant amount of energy, its performance mostly depends on the ability of the clustering phase to cluster the network efficiently [19]. During each tour, the AUV visits all CHs to collect the aggregated data and then returns to the static sink to deliver all received data and starts the next data gathering tour.

B. PROBLEM COMPLEXITY

Our main objective is to find a subset of sensors as CHs in order to minimise the length of the tour going through them while other sensors are covered within the relay hop bound of those CHs. Before giving the formulation, we discuss the computational complexity of this problem. We have the following theorem regarding our mobile data gathering problem called CMDG.

Theorem 1. *CMDG problem belongs to the class of NP-hard problems.*

Proof. All sensors can become unreachable from each other if the transmission range of sensors is reduced to a certain level. In this case, the Travelling Salesman Problem (TSP) which is NP-hard can be reduced to a special case of CMDG problem in polynomial time (i.e. $TSP \leq_P CMDG$). Let TR be the transmission range of sensors. The AUV can collect data from a sensor when it enters the transmission of the

sensor. Note that when TR is below a certain level (i.e. $TR \rightarrow 0$), the AUV should find a minimal tour visiting all sensors which is known as TSP problem. In this case, the TSP problem is reduced to our problem (CMDG) in polynomial time. Thus, the CMDG problem is NP-hard. \square

C. PROBLEM FORMULATION USING MIP

In this section, we formulate the mobile data gathering scheme into an optimization problem. Let $S = \{s_0, s_1, \dots, s_n\}$ be a set of n sensors randomly distributed in the region of interest with one sink node, s_0 . The cost of the distance between sensors s_i and s_j is also represented by $q_{i,j}$. We also assume $q_{i,i} = \infty, \forall i \in \{0, \dots, n\}$. The d -hop neighbours of each sensor, s_i , is indicated by $N_d(s_i)$. In a formal way, our problem is to find a subset of S to form the cluster heads in which every other member of S is covered by at least one CH with a maximum number of d hops, and a tour with the minimum length of T can be determined to visit all CHs.

We define the following decision variables:

- $z_{i,j}$ is a binary variable which is one if tour contains the link (s_i, s_j) .
- $f_{i,j}$ indicates the flow on link (s_i, s_j) ; note that $f_{i,j} = 0$ when $z_{i,j} = 0$.
- u_i is a binary variable which is one if tour includes node s_i as a CH.

Our problem is formulated as a Mixed-Integer Programming (MIP) as follows:

Find variables $z_{i,j}, f_{i,j}, u_i$ which satisfy:

$$T = \min \left\{ \sum_{i=0}^n \sum_{j=0}^n q_{i,j} z_{i,j} \right\} \quad (1)$$

$$\text{s.t. } \sum_{j \in N_d(s_i)} u_j \geq 1, \quad j = 0, 1, \dots, n, \quad (2)$$

$$\sum_{j=0}^n z_{i,j} = u_i, \quad i = 0, 1, \dots, n, \quad (3)$$

$$\sum_{i=0}^n z_{i,j} = u_j, \quad j = 0, 1, \dots, n, \quad (4)$$

$$\sum_{i=1}^n f_{i,j} - \sum_{k=1}^n f_{k,j} = u_j, \quad j = 1, \dots, n, \quad (5)$$

$$\sum_{j=1}^n f_{j,0} = \sum_{j=1}^n u_j, \quad (6)$$

$$f_{i,j} \leq |S| z_{i,j}, \quad i = 0, 1, \dots, n, \quad j = 0, 1, \dots, n. \quad (7)$$

Letting $z_{i,j}^*, f_{i,j}^*, u_i^*$ be the optimal solution to MIP problem, the graph $G(\Phi^*, E)$ can be associated with that solution, where Φ^* is the CHs set and $E = \{(i,j) | z_{i,j}^* = 1\}$ is equal to its arc set.

In the above formulation, the objective function (1) minimises the tour length T . Constraint (2) ensures that each sensor can be covered by at least one CH within the d -hops distance, such that its packets can be collected dur-

ing the AUV tour. The coverage of each sensor depends on the underwater physical layer model which is defined in Section III-B. Thus, d -hop connectivity and coverage in the tour planning formulation are determined by parameters like the transmission power, packet size, channel attenuation and noises [4], [38]. Constraints (3-4) guarantee that the AUV enters and departs each selected CH and the data sink only once. Constraints (5)-(7) are considered to exclude the solutions with sub-tours and those without including sink as the starting and ending point. Constraint (5) ensures that there is an incoming and outgoing flow for each selected CH, while outgoing flow is one unit more than the incoming flow. Constraint (6) restricts that the incoming flow to the sink node, s_0 , is equal to the number of selected CHs in the tour. Finally, constraint (7) enforces that link (s_i, s_j) can only have a flow if it is included in the tour.

Our problem formulation can also be extended for multiple sinks. In this case, each sink is considered as an isolated node which should be visited during the AUV tour. The constraint (2) guarantees that these sink nodes are included in the tour. The impact of multiple static sinks on mobile data gathering performance depends on different parameters such as the location of static sinks, network dimensions and density [33].

To perform the optimal solution, the d -hop degree and geographical coordinates of each sensor are required. The d -hop degree of each sensor can be calculated by having one-hop adjacency information and constructing the neighbourhood graph. The sensors geographical coordinates are also used to calculate the pairwise distances between sensors. This information can be gathered during the discovery phase. In the discovery phase, each underwater sensor broadcasts a control packet including the packet type and sensor ID. Upon receiving a control packet, each receiving sensor updates its neighbouring table based on the newly discovered sensor.

After the information exchange, the AUV needs to travel the entire sensing field to collect sensors information and marks their locations. Underwater sensors are unaware of their location; however, the AUV can obtain sensors co-ordinates by marking the locations where it receives data from them. While exploring, the AUV can broadcast control packets periodically to discover the sensors. When a sensor receives a control packet from the AUV, it should respond to AUV using an ACK message. The AUV then marks its current location and links it with the ID of the sensor.

After exploring of all sensors, the AUV returns to the static sink to upload the sensor list and their locations. Using the collected information, the monitoring centre solves the optimisation problem to determine the cluster heads, members and the AUV tour. After the tour planning, the clustering information should be disseminated in the entire field by AUV. The AUV data gathering can be initiated after the tour planning and data dissemination.

If the network topology is static, the planned tour and clusters are fixed over time. However, if the network is dynamic, a mobility handling mechanism is required to maintain the data gathering performance and keep the initial clusters valid

Algorithm 1 Centralised Algorithm

```

1: procedure Centralised( $S, maxitr, \Phi$ )
2:  $numCH = \infty$ 
3:  $totalDist = \infty$ 
4: for  $k = 1$  to  $maxitr$  do
5:   Select  $\lambda$  randomly from interval  $[0, 1]$ 
6:   Clustering( $S, \lambda, \Phi$ )
7:   for each  $s_i \in \Phi$  do
8:      $dist = dist + Distance(s_i, Sink)$ 
9:   end for
10:  if  $|\Phi| < numCH$  and  $dist < totalDist$  then
11:     $numCH = |\Phi|$ 
12:     $totalDist = dist$ 
13:     $\Phi^* \leftarrow \Phi$ 
14:  end if
15: end for
16: end procedure

```

Algorithm 2 Clustering Algorithm

```

1: procedure Clustering( $S, \lambda, \Phi$ )
2:  $\Phi \leftarrow \emptyset$ 
3:  $C \leftarrow S$ 
4:  $l = 0$ 
5: while  $|C| > 0$  do
6:    $G(C) \leftarrow$  Sub-graph induced by  $C$ 
7:   for each  $s_i \in C$  do
8:     Calculate  $d\_deg(s_i)$  with respect to  $G(C)$ 
9:   end for
10:   $d_{min} = \min\{d\_deg(s_i) \mid s_i \in C\}$ 
11:   $d_{max} = \max\{d\_deg(s_i) \mid s_i \in C\}$ 
12:   $RCS = \{s_i \in C \mid d\_deg(s_i) \geq d_{min} + \lambda(d_{max} - d_{min})\}$ 
13:  Select  $s_i$  at random from the  $RCS$ 
14:  Remove  $s_i$  and  $d\_neigh(s_i)$  from  $C$ 
15:   $l = l + 1$ 
16:   $\Phi(l) \leftarrow s_i$ 
17: end while
18: end procedure

```

for a longer time. The mobility handling mechanism will be discussed in Section V.

D. CENTRALISED CLUSTERING ALGORITHM

A new centralised clustering algorithm is proposed in this section to obtain local optimal solutions for the CMDG clustering problem. In order to solve the problem approximately, CMDG is performed iteratively until an acceptable solution can be found in a limited amount of time. The features of randomness and greediness are simultaneously used in the CMDG clustering algorithm to generate different solutions in each local search while providing a faster convergence to a local minimum.

Some effective features should be determined to find the optimal solution among all the solutions found in the cen-

TABLE 1: Notations used in Algorithms 1-3

Symbol	Definition
S	Sensors list
$maxitr$	The maximum number of iterations
Φ	Temporary list to maintain the selected CHs
Φ^*	List of selected CHs
$numCH$	Number of CHs
$totalDist$	The total distance of all selected CHs to the sink
λ	A random value within $[0, 1]$
$dist$	A temporary variable to calculate the distance
$Sink$	The static sink
C	The set of candidate sensors
$G(C)$	A sub-graph induced by C
l	A temporary variable to calculate the number of CHs
$d_deg(s_i)$	The d -hop degree of each candidate sensor s_i
d_{min}	The minimum d -hop degree of $G(C)$
d_{max}	The maximum d -hop degree of $G(C)$
RCS	The restricted candidate set
$d_neigh(s_i)$	The d -hop neighbours of sensor s_i
t	A delay timer based on d -hop degree of each sensor
δ_d	The sensor d -hop degree
Δ_d	The maximum d -hop degree in the network
λ	A short random time duration
DP	The distributed clustering time interval
CH	Cluster head
non_CH	None cluster head status
CH_msg	Cluster head announcement message
$join_msg$	The join request message

tralised algorithm. First, the distance of the selected cluster heads to the sink node should be taken into consideration. Cluster heads distributed close to the sink usually lead to having a shorter tour. Second, having a smaller number of CHs can contribute to having a shorter tour as well. The CMDG centralised clustering is proposed in Algorithm 1. Table 1 also shows the notations used in this paper.

Let $S = \{s_1, s_2, \dots, s_n\}$ be a set of acoustic sensors which are uniformly distributed in the region of interest. The temporary set, Φ , is used to maintain the selected CHs of each solution. After each iteration, the new solution is compared with the best solution found so far, and may replace it if obtained a better result. The $numCH$ indicates the number of CHs, and $totalDist$ keeps the total distance of all those CHs to the sink. The initial values of $numCH$ and $totalDist$ are set as infinite (lines 2-3).

In order to obtain a wider range of solutions, the maximum number of iterations, $maxitr$, should be properly selected (line 4). At each iteration, a random value is selected from interval $[0, 1]$ and assigned to λ as a real parameter (line 5). A new greedily biased solution is then generated by calling the clustering procedure. In this procedure, a group of sensors is selected as CHs while each of the remaining sensors is associated with one CH following the relay hop bound limitation (line 6).

The solution Φ obtained from the clustering procedure is compared with the best solution Φ^* found so far. The total distances of all CHs within Φ to the static sink should be calculated and assigned to the variable $dist$ (lines 7-9). If $dist$ is less than the $totalDist$ and the number of CHs is less

than the $numCH$, the current solution Φ is stored as a better solution in Φ^* , and it replaces $totalDist$ and $numCH$ with the values of $dist$ and $|\Phi|$, respectively (lines 10-13).

The clustering procedure is presented in Algorithm 2. Let C be the set of candidate sensors which includes all sensors at the beginning and Φ is initially set to empty (lines 2-3). The main idea is to select a sensor randomly from the candidate sensors with the high d -hop degree value, adding the selected sensor to Φ , and removing the selected sensor with its d -hop neighbouring sensors from the candidate list. The number of CHs is calculated and maintained in the variable l (line 4).

The clustering procedure proceeds until C becomes empty (line 5). At each iteration, a sub-graph induced by C is built and assigned to $G(C)$. The sub-graph $G(C)$ is used to recalculate the d -hop degree of each candidate sensor (lines 6-9). The minimum and maximum d -hop degree of $G(C)$ are shown by d_{min} and d_{max} , respectively (lines 10-11). The Restricted Candidate Set (RCS) is defined to maintain the candidate sensors with the higher d -hop degree value than $d_{min} + \lambda(d_{max} - d_{min})$. One sensor is randomly picked from RCS to be included in Φ . The selected sensor with its d -hop neighbouring sensors should be excluded from C (lines 12-16). The procedure continues as long as C is not empty. At the end, Φ is returned as a solution to the main function (Algorithm 1) where its optimality is compared with the best solution found so far. After finding the CHs of the best solution, each non-CH sensor can join the closest CH in the neighbourhood.

The list of selected CHs of an optimal local solution is given by Φ^* . The visiting order of CHs within Φ^* should be determined to obtain the shortest tour. Finding such a visiting order is a well-known TSP problem which is discussed in detail in Section IV-F. After the tour construction, all sensors are informed by AUV about the selected CHs and members.

E. DISTRIBUTED CLUSTERING ALGORITHM

In practice, the centralised scheme may need global information which is difficult to obtain. Thus, in this section, a novel distributed algorithm is proposed to determine the clusters with the aim of creating a short tour without using any localisation information. A shorter tour for the AUV can usually be obtained by having a lower number of clusters with higher density. Therefore, the distributed clustering scheme gives priority to a sensor with a higher d -hop degree to become a CH. The distributed clustering phase is described in Algorithm 3.

Before the distributed clustering phase, each sensor is required to obtain and maintain d -hop neighbouring information including the number of neighbours and pairwise distances. During the distributed clustering, there is a competition between sensors to become a CH based on the d -hop degree. A sensor with the ability to cover more sensors has a higher chance to be a CH than others. At the end of the distributed clustering phase, each non-CH sensor may have received declaration messages from different clusters and it should join the closest CH by sending a join message.

Algorithm 3 Distributed Clustering Algorithm

```

1: procedure Clustering(sensor)
2: Set a delay timer  $t = (\Delta_d - \delta_d) \times (DP/\Delta_d) \pm \lambda$ 
3: while the delay timer is not expired do
4:   Listen and maintain the  $CH\_msgs$ 
5:   if  $CH\_msg$  has been forwarded for less than  $d$ -hop then
6:     Forward the  $CH\_msg$ 
7:   end while
8:   if never received any  $CH\_msg$  then
9:      $status(sensor) \leftarrow CH$ 
10:    Broadcast a  $CH\_msg$ 
11:   else
12:    Keep listening and maintaining the  $CH\_msgs$ 
13:   end if
14:   if  $DP$  is over and  $status(sensor) \neq CH$  then
15:      $status(sensor) \leftarrow non\_CH$ 
16:      $CH(sensor) \leftarrow$  the closest  $CH$  among accessible  $CHs$ 
17:     Transmit a  $join\_msg$ 
18:   end if
19: end procedure

```

A timer-based approach is used to determine the CHs. Upon starting the clustering phase, each sensor sets a delay timer which has an inverse relation with its d -hop degree (line 2). In this way, a sensor with a higher number of neighbouring nodes in its d -hop range is delayed for a shorter time and consequently has a higher chance to become a CH. Each sensor computes its delay timer based on its d -hop degree, δ_d , as the following

$$t = (\Delta_d - \delta_d) \times (DP/\Delta_d) \pm \lambda \quad (8)$$

where DP is the clustering time interval and Δ_d indicates the maximum d -hop degree of a network topology which can be determined based on the network density, deployment strategy, and network dimensions. Each sensor knows the Δ_d value during the deployment time. A short random time duration, λ , is also considered to differentiate the sensors with the same δ_d .

Before the delay timer is expired, a sensor may receive the CH declaration messages, CH_msgs , from its d -hop neighbouring nodes with higher priority. The receiving sensor should forward the received CH_msg to its neighbours if it has been propagated for less than d hops (lines 3-7).

After the delay timer is expired, a sensor becomes a CH if no CH_msg received from other d -hop neighbouring sensors during the delay period. The CH should broadcast a CH_msg to be received by all d -hop neighbouring sensors. A non-CH sensor should also listen and maintain the CH_msgs until the end of the clustering period (lines 8-13).

At the end of the distributed clustering period, a non-CH sensor should send a joining message, $join_msg$, to a CH with the closest distance. In this way, all sensors are clustered while their relay hop is bounded to d -hops (lines 14-18). All selected CHs should be discovered and their locations are

marked by the AUV. The list of CHs is then uploaded to the static sink for the tour planning [19].

F. DATA GATHERING PHASE

Finding the shortest possible route to visit each CH in the field and returning to the starting point is a well-known problem called Travelling Salesman Problem (TSP) [43]. As TSP belongs to the class of NP-hard problems, heuristic algorithms are considered to obtain the near-optimal solution in less computational time. Therefore, in this section, we use a greedy heuristic algorithm to find a short path passing through all CHs obtained from the clustering phase [43].

Let N be the number of cluster heads in the field. Assume there is an edge between each pair of CHs indicating the geographical distance between them. In the greedy heuristic algorithm, edges are sorted in the ascending order and being added to the tour, from shortest to longest, as long as there is no CH with a degree more than 2, and no early cycle is formed before including N edges. The following algorithm summarises the procedure:

1. Edges between CHs should be sorted in the ascending order of length.
2. Add the shortest edge to the tour as long as it does not
 - Form an early cycle, and
 - Create a CH with a degree more than two
3. If the tour constructed so far does not include N edges, step 2 is repeated.

The greedy heuristic algorithm has the complexity of $O(n^2 \log_2(n))$ where n is the number of CHs [43].

The tour constructed by the greedy heuristic is not optimal, as some of its edges may cross each other. The 2-opt algorithm can be used to further optimise the initial tour by converting it to a tour without any crossed line [44]. In this algorithm, the initial tour is incrementally improved by swapping all possible pairs of edges, and replacing the current tour with the new tour if it shortens the tour length. This procedure is continued until no more improvements can be made. The 2-opt has the complexity of $O(n^2)$ [45].

The AUV data gathering should be started after the tour planning. The planned tour is used by AUV to visit each CH and collect the aggregated data. Each CH also buffers the data packets received from its members and transfers them to the AUV when it arrives. Members transmit data packets to their CHs with a fixed data rate which is independent of the AUV activity.

When approaching a CH, the AUV transmits a control packet to announce its arrival. Each CH then starts sending the data packets to AUV after notifying the members to be suspended during this period. Then, the AUV travels to the next CH and repeats the same procedure until it returns to the static sink. The next round of data gathering is similarly repeated.

V. DISCUSSION ON THE SENSORS MOBILITY

For the completeness of the routing algorithm, sensor mobility should also be addressed. In UWSNs, two types of

network may be used: static and dynamic. In a static network, sensors are anchored to the bottom of the sea and are fixed during the entire network lifetime. In a dynamic network, sensors can move with the water current. In this section, we propose a mechanism to make our scheme resistant to sensor mobility.

The established routing paths may become invalid over the time while sensors continuously move with the water currents. However, sensor mobility in a marine environment has its own specific characteristics. Underwater sensors continuously move in the horizontal direction; however, vertical displacements are almost negligible with respect to the horizontal ones. Furthermore, sensors usually move together at a low speed with the same pattern. Thus, a realistic mobility model gives a better view of the impact of sensors mobility on an underwater routing protocol.

A. SENSOR MOBILITY MODEL

The Meandering Current Mobility (MCM) model is used to capture the physical movement of the sensor nodes [46]. By using the stream-function ψ , the trajectory of a sensor located at (x, y) can be calculated as

$$\dot{x} = -\frac{\partial \psi(x, y, t)}{\partial y}; \quad \dot{y} = \frac{\partial \psi(x, y, t)}{\partial x} \quad (9)$$

where \dot{x} is the zonal (eastward) and \dot{y} is the meridional (northward) drift of the sensor in time t . The stream-function which represents a jet-like current is given by [46]

$$\psi(x, y, t) = -\tanh \left[\frac{y - B(t) \sin(k(x - ct))}{\sqrt{1 + k^2 B^2(t) \cos^2(k(x - ct))}} \right] \quad (10)$$

where c and k are the phase speed, and wave-number, respectively. The function $B(t)$ indicates the amplitude of the meanders which is given by

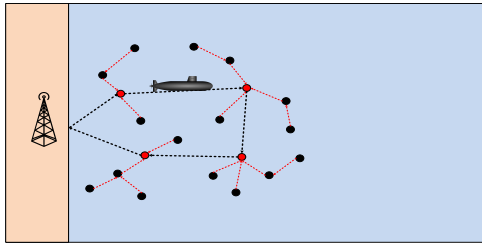
$$B(t) = B_0 + \varepsilon \cos(\omega t) \quad (11)$$

where B_0 is the average meander width. The parameters ε and ω determine the amplitude and frequency of the modulation, respectively.

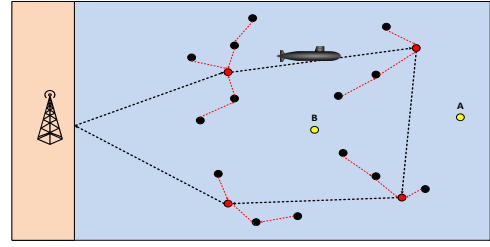
B. MOBILITY HANDLING MECHANISM

To maintain the routing performance in dynamic environments, we can add a straightforward technique to our proposed scheme. In this way, the initial clustering stays valid for a longer time. During the data gathering phase, the AUV only needs to track and update the position of CHs, and members may join different CHs during this phase. Motivated by the fact that sensors nearly move together, we assume that most of the members remain connected to their CH and some may join to a new CH when they lose any bounded hop connection with the previous CH.

Upon joining a cluster, a sensor sets a validation timer that its value should be greater than the packet sending interval. During this time, if the sensor senses any packet transmission



(a) The initial AUV tour constructed by CMDG schemes



(b) The AUV tour status after sensors have moved with the water current

FIGURE 4: The AUV tour status in the presence of mobility

from its CH or other members, it only updates its connection accordingly and resets the validation timer. Otherwise, if no neighbouring sensor within the same cluster is sensed before the expiration of the validation timer, the sensor looks at other neighbouring sensors to check whether it can connect to a new cluster in compliance with the hop-bound requirement. If yes, the sensor transmits a joining message to the CH with the closest distance. If no, sensor probably has no neighbour or it cannot join to any cluster because of the violation of hop-bound constraints. An isolated sensor should wait until it can join a new cluster. Using this technique, the updating and clustering phases do not need to be repeated, when the network topology updates.

Assuming the mobility has been presented to the system model, Fig. 4a shows an example of AUV tour at the beginning of its operation and Fig. 4b shows its status after a while. It is obvious that the tour length increases if sensors have drifted away from the sink. The members of each cluster may have been changed during the operation time, and some sensors like *A* and *B* have become isolated. This is because sensor *A* can see no neighbouring sensor in its vicinity and sensor *B*, despite having the neighbouring sensors, cannot join a cluster without violation of the hop bound limitation (2-hop bound in this example). The impact of sensor mobility on the tour length and network connectivity will be discussed in Section VI-B.

VI. SIMULATION RESULTS

In this section, the simulation setup and also the experimental results are proposed. First, simulation is conducted to evaluate the impact of sensor mobility on the tour length and network connectivity. The CMDG performance is then assessed against that of the optimal solution in a small-scale network. Finally, extensive simulations are conducted to evaluate the performance of centralised and distributed CMDG (CMDG-cent and CMDG-dist) against those of recently proposed mobile data gathering schemes, *AUV_PN* [34], and *AEERP* [32].

A. PERFORMANCE METRICS

The following metrics are defined to evaluate the proposed algorithms.

Packet Delivery Ratio (PDR): It indicates the ratio of the number of packets successfully received by the sink to the number of packets generated by the sensors.

Energy tax: It is defined as the average energy consumed per packet to successfully be delivered to the static sink.

End-to-end delay: It shows the average time for data packets to being delivered from sensors to the static sink.

Tour length: It indicates the total distance travelled by AUV in each data collection round.

Relay hop count: It is defined as the average number of hops a packet relayed from sensors to their CHs.

Number of cluster heads: It is defined as the number of cluster heads that the AUV should visit them during the data collection phase.

Isolated sensors: It shows the number of sensors which have no neighbouring sensor or those who cannot join any cluster without violating the hop-bound limitation.

Network lifetime: It is defined as the time when the first sensor runs out of energy. Normalized network lifetime is obtained by dividing all data points in a figure by the largest value [47], [48].

B. PERFORMANCE EVALUATION WITH SENSOR MOBILITY

In this section, we evaluate the impact of sensor mobility on the tour length and network connectivity over time. The MCM model described in Section V is used in our simulation. We also evaluate our scheme using a Random Walk 2D mobility model (moving in the X-Y plane) in which each sensor moves independently from the others. In the simulation, the tour is initially constructed using a CMDG-dist ($d = 2$) and then sensors move with either of these mobility models.

1) Simulation setup

We consider a 3D underwater network with 200 sensors which are initially deployed in a two-dimensional plane $1000m \times 1000m$ at a fixed depth. We use a transmission range equal to 100 meters for each sensor. The static sink is placed on the surface at (0, 500, 0) coordinates. We use the stream-function in Eq. (10) with the same time and space scalings used in [46] as representative of a MCM mobility model in our simulation. We also use $B_0 = 1.2$, $\varepsilon = 0.3$, $\omega = 0.4$, $c = 0.12$, and $k = 2\pi/7.5$ in Eqs. (10) and (11).

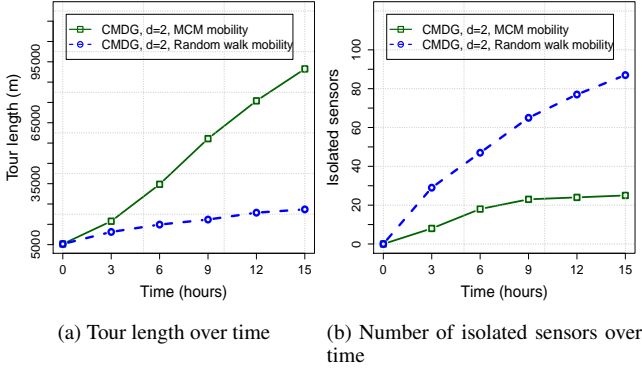


FIGURE 5: Performance evaluation with sensor mobility

In random walk mobility model, sensors move in random directions with 0.3 m/s speed. All the results are averaged over 50 runs for randomly generated topologies while the simulation time is ranging from 0 to 15 hours.

2) Results

The simulation results help us to understand when it is necessary to repeat the clustering phase based on the application requirements when the mobility is random or predictable. Figs. 5a and 5b show the tour length and the number of isolated sensors over time using different mobility models. As can be seen in Fig. 5a, the tour length increases over time in under both mobility pattern.

There is a more considerable increase in the tour length using the MCM mobility model. This is due to the fact that underwater sensors are continually being drifted away from the sink by the water currents. However, the number of isolated sensors in MCM model increases slowly. This is because the adjacent sensors usually move together with the same pattern. Thus, the pre-built clusters can stay valid for a longer time and also a sensor separated from a cluster has more chance to join another cluster in its vicinity.

The increase in the tour length in the random walk model is less because sensors can move randomly in any direction and consequently their dispersion distance to the sink is less. However, the number of isolated sensors increases very fast because sensors move independently from each other and therefore more CHs and members become separated.

C. PERFORMANCE COMPARISON WITH OPTIMAL SOLUTION

In this section, we compare the results of CMDG, both centralised and distributed, with the optimal solution for small networks. We use CPLEX [49] to obtain the optimal solution for the MIP problem in Section IV-C.

1) Simulation setup

We consider that acoustic sensors, varies from 10 to 40, are uniformly distributed in a two-dimensional plane $500\text{m} \times$

500m at a fixed depth. The static sink is located on the surface at point $(0, 250, 0)$. Each sensor has a transmission range of 100 meters. The relay hop bound, d , is set to 2 for the proposed schemes and the optimal solution. All the results are averaged over 50 runs for randomly generated topologies.

2) Results

Fig. 6a shows that the tour length is increased when sensors are sparsely deployed (10 to 25 sensors). This is because CHs are selected from a small number of sensors which makes them dispersed from each other and the static sink. The tour length is decreased when the network is dense (25 to 40 sensors). This is because CHs can be selected in the proximity of each other and with less dispersion from the static sink. The CMDG-cent algorithm outperforms CMDG-dist to find a shorter tour and close to the optimal solution utilising the global information to select the CHs.

Fig. 6b indicates that the number of CHs is increased when the number of sensors increases from 10 to 25. This is because a greater number of sensors has a chance to become a CH when the network is still sparse. However, when the network is dense, a fewer number of CHs is required to cover the network field. The lowest number of CHs is used by CMDG-cent because its primary criterion is having the minimum number of CHs.

As can be observed from Fig. 6c, the average relay hop count is increased by increasing the number of sensors. This is because more sensors can be located in the d -hop distance of CHs when a higher number of sensors are deployed. For CMDG-dist, its tour may be longer than others; however, its average relay hop count is less than other schemes which can contribute to more energy saving.

D. PERFORMANCE COMPARISON WITH OTHER PROTOCOLS

In this section, the CMDG performance is evaluated against other mobile data collection schemes recently reported in the literature, AEERP and AUV_PN, in terms of the packet delivery ratio, energy tax, tour length, end-to-end delay, and network lifetime.

1) Simulation setup

In our simulation, we use the underwater communication channel described in Section III. The transmission power is set to $105 \text{ dB re } \mu \text{ Pa}$, and the power level threshold for receiving packets without errors is $10 \text{ dB re } \mu \text{ Pa}$. The bit rate is considered as 10 kbps .

We use CSMA MAC protocol with retransmission capability to handling the packet errors and also offsetting the effects of high propagation delay in the underwater environment [50]. When a sensor senses a free channel, it forwards the data packet; otherwise, a back-off algorithm is invoked. After three times back-off, the forwarding node discards the data packet. Upon the packet reception, the receiving sensor sends back a short ACK packet. If the forwarding node fails to hear

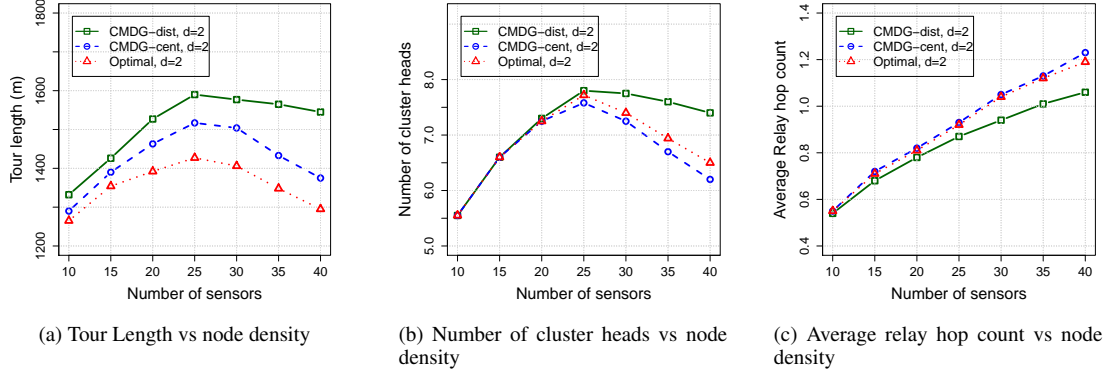


FIGURE 6: Performance comparison with optimal solution

an ACK packet, the data packet should be retransmitted. The packet is dropped after three retransmissions.

It is considered that sensors (ranging from 100 to 500) are uniformly deployed in a $1000m \times 1000m$ field with a fixed depth at 300 m. Each sensor has a transmission range of 100 meters. In order to send and receive a data packet, 50 W and 0.158 W energy are consumed, respectively. A sensor consumes 0.008 W energy in the idle mode.

Each sensor sends a data packet with the size of 1024 bits at every 100 seconds. The AUV speed is equal to 4 m/s while it travels at a depth of 250 m. The static sink is considered at the corner of the network topology with (0, 500, 0) coordinates. The distributed clustering phase is considered as 80 seconds and the Δ_d in Eq. (8) is set to 40. All the results are averaged over 50 runs for randomly generated topologies while the simulation time for each run is set to 12 hours. In order to evaluate the network lifetime, we let the simulation run until the first node runs out of the energy. The initial energy for each node is taken as $E_0 = 40000J$.

2) The impact of relay hop bound

In this set of simulations, we evaluate the performance of CMDG-cent and CMDG-dist as a function of d . We fix the number of sensors at 200, and other system parameters are the same as those described in Section VI-D1.

From the Figs. 7a and 7b, it can be seen that by increasing d , the tour length is shortened and consequently the average end-to-end delay decreases because each AUV tour is completed faster by travelling in a shorter tour, and data packets are collected at a faster rate. CMDG-cent outperforms CMDG-dist in finding a shorter tour and obtaining a lower end-to-end delay for packet delivery. It is because CMDG-cent has a general view and it looks for a near-optimal solution by generating different random and greedy solutions; however, CMDG-dist can only generate one solution which is limited to a timer-based clustering.

Fig. 7c shows that the number of total CHs drops as d increases. The reduction in the number of CHs in CMDG-cent is more significant because its primary criterion is to

cluster the sensors with the minimum number of CHs. On the other hand, as can be seen in Fig. 7d, the average relay hop count increases as d becomes larger because when d is high, more sensors within a higher average hop count distance can join a CH.

An increase in the average relay hop count results in a reduction in the packet delivery ratio and an increase in energy consumption, as shown in Figs. 7e and 7f, respectively. This is due to the fact that the packet failure probability and also the energy consumption increases when the packets must be relayed in a higher number of hops to the CH on average. Although CMDG-dist has a larger tour length, it can obtain a higher packet delivery ratio and lower energy consumption due to having more clusters with less average hop count distances to each CH.

3) The impact of sensor density

In this set of simulation scenarios, the impact of sensor density on the performance of CMDG, AUV_PN, and AEERP are examined. The number of sensors varies from 50 to 500 while other parameters are fixed (as described in Section VI-D1). The simulation results for the tour length, number of cluster heads, packet delivery ratio, energy tax, and network lifetime are shown in Figs. 8a, 8b, 8c, 8d, and 8e, respectively.

From Fig. 8a, it can be observed that the tour length of AUV_PN is higher than CMDG schemes. In AUV_PN, AUV initiates the data gathering tour from a static base station and moves to the nearest CH to retrieve the list of Path Nodes (PNs). After obtaining the PNs list, the AUV travels to each PN to gather data packets. However, the tour constructed by AUV_PN is not optimal as it crosses over itself. Nevertheless, the tour established by CMDG scheme is optimised using the 2-opt algorithm. In AEERP, the AUV travels a short path which is fixed; however, it increases the energy consumption and packet failure as can be observed in Figs. 8c and 8d.

The number of CHs in different network densities is plotted in Fig. 8b. In CMDG schemes, an increase in the number of CHs can be observed when the number of sensors is increased from 50 to 100 sensors. This is due to the fact that

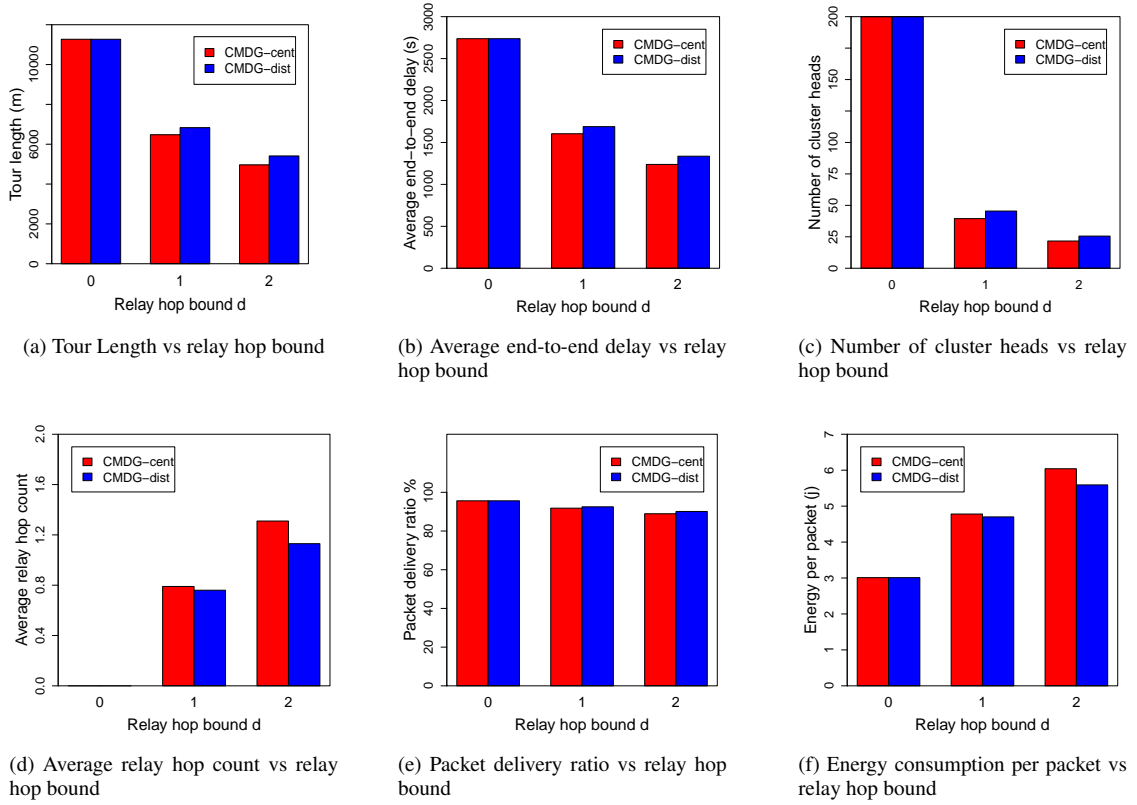


FIGURE 7: The impact of relay hop bound on CMDG

with the fewer number of sensors, the network is not well-connected. However, by having more sensors available, the network topology becomes dense and fully connected. The network is then can be covered with the less number of CHs. Nonetheless, AEERP has a higher number of CHs within a dense network since a greater number of sensors can be located close to the elliptical path, increasing their chances to become a CH. Eventually, it is observable that the number of CHs (including PNs) is fixed in AUV_PN.

Fig. 8c plots the packet delivery ratio in different network densities. CMDG schemes have a higher packet delivery ratio compared to those of other schemes because of bounding the packet relay to $d = 2$ hops. However, AUV_PN does not consider any relay hop bound on the number of packet transmissions. Thus, the packet failure rate increases by increasing the distance between members and PNs and consequently transmitting a packet over a longer range. The packet failure probability in AEERP is very high because the void area may occur between some members and the associated CH when the network is not well-connected. There is also no relay hop bound between members and CHs. Therefore, AEERP has the lowest packet delivery ratio in sparse scenarios.

Fig. 8d plots the average energy consumed per packet in each scheme. The energy consumption of CMDG-dist is significantly less than those of others. It is due to the fact

that the number of transmissions is considerably decreased by considering the relay hop bound.

The AUV_PN consumes higher energy than CMDG-dist. This is because, in AUV_PN, the number of CHs is less than CMDG-dist. The fewer number of CHs results in a higher distance between sensors and CHs and therefore wasting more energy because of packet transmission over a longer range. Furthermore, in AUV_PN, cluster heads are not engaged in data collection, and they are only visited by AUV for obtaining the list of PNs. Therefore, the number of CHs participating in local data aggregation is less than the actual value.

AEERP wastes a lot of energy in a sparse network because of the void problem. When the network is dense, the network becomes well-connected and the void area is mostly disappeared. However, AEERP still consumes a higher amount of energy because of having no relay hop bound constraint.

Fig. 8e shows that the network lifetime is decreased when the network density increases. This is because a higher number of sensors are assigned to a cluster in a dense network resulted in rapid energy depletion of CHs. The network lifetime of CMDG-dist is higher than those of others because sensors are uniformly assigned to a large number of clusters. In AEERP, the number of CHs is higher than CMDG-dist when the network is dense; however, the sensors are not

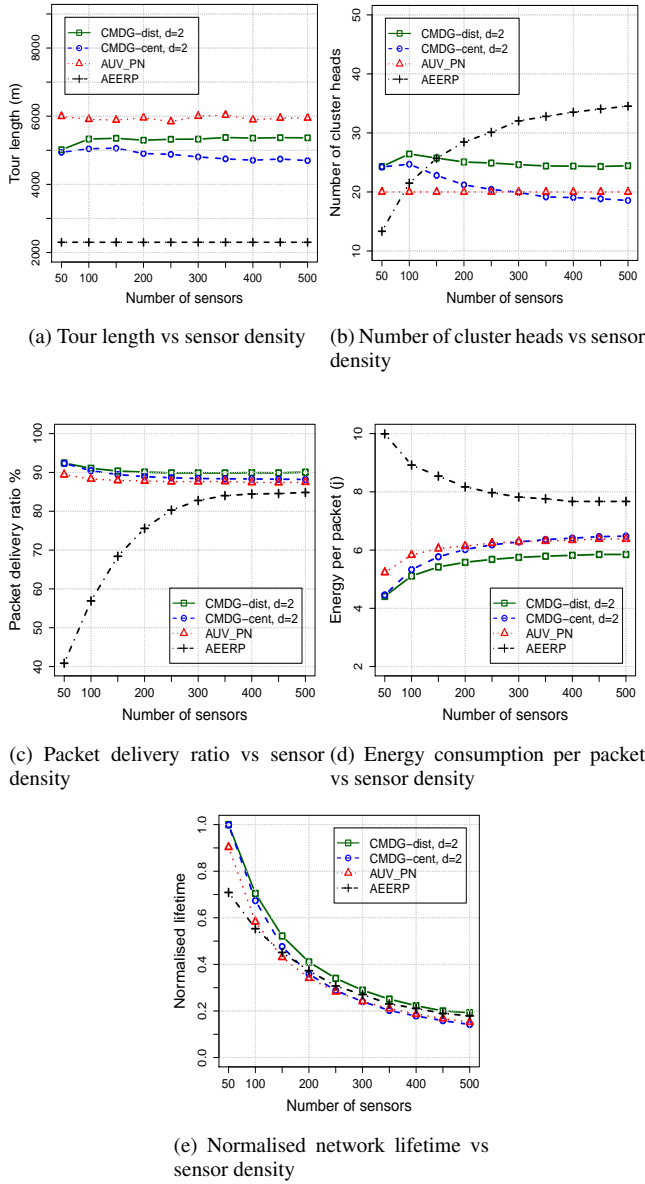


FIGURE 8: The impact of sensor density on the mobile data gathering schemes

uniformly assigned to the CHs which can deplete the energy of a group of CHs very quickly resulted in a shorter lifetime than CMDG-dist. The lower number of clusters in AUV_PN also keeps its lifetime lower than CMDG-dist.

4) The impact of network area size

In this section, the impact of network area size on the proposed schemes is investigated. The number of sensors is fixed at 200 while the side length of the distributed area, SL , is varied from 500 m to 2000 m. In AEERP, the semi-major and semi-minor axes of the elliptic path are set to $SL/2$ and $SL/5$, respectively. Figs. 9a, 9b, 9c, 9d, and 9e plot the results for the tour length, number of cluster heads, packet delivery

ratio, energy tax, and network lifetime, respectively.

Fig. 9a indicates that the tour length increases as SL is increased. This is due to the fact that an AUV should travel a longer tour to visit all clusters formed in a wider area. In a small area scenario, AUV_PN has the highest tour length (SL is less than 1000m). However, in a broader area, AUV_PN has a shorter tour compared to those of CMDG schemes. This is because the geographical distances between CHs are increased while the number of CHs is fixed in AUV_PN (as shown in Fig. 9b). On the other hand, CMDG schemes use a higher number of CHs to maintain the relay hop bound and cover more positions in a wider area. Therefore, the tour length is increased as an expense while the packet delivery ratio and energy saving are considerably improved as shown in Figs. 9c and 9d, respectively.

Fig. 9c shows that the packet delivery ratio in CMDG schemes increases as SL becomes longer. This is because the number of CHs in CMDG schemes is increased by increasing SL as shown in Fig. 9b, and consequently the average relay hop count decreases. By decreasing the average relay hop count, the packet delivery probability increases as each packet should be relayed over a lower number of hops on average. In AEERP, when SL is small, the network is well-connected with possibly very few void areas between members and CHs, and consequently its packet delivery ratio is high. However, when SL increases, the hop distance between members and CHs is increased and the chance of packet failure is increased because of the higher chance of void occurrence or high bit error rate. The packet delivery ratio in AUV_PN also drops as SL increases because the number of CHs remains fixed and only the distance of members to their CHs is increased. The higher distance between the members and CHs can increase the packet failure probability.

As shown in Fig. 9d, the energy consumption of AUV_PN increases as SL is increased. In AUV_PN, by increasing SL , the distance between sensors and CHs is increased while the number of CHs is still fixed. Therefore, a data packet needs to be transmitted with a higher power to cope with the attenuation and path loss over a long distance. The energy of AEERP is also increased as SL increases because packets should be relayed over more hops while some of them fail because of the existence of void communication area. However, the energy consumption of CMDG schemes is decreased as SL increases. This is because, in a wider area, the number of CHs increases while the number of sensors is fixed. Therefore, sensors are allocated to each CH with lower hop count distances resulted in more energy saving.

Fig. 9e shows the normalised network lifetime as a function of SL . The network lifetime of CMDG schemes is increased as SL increases. This is because the number of clusters is increased as the sensors are distributed in a wider area. Thus, the lower number of sensors are assigned to each CH resulted in a higher network lifetime. However, in a broader area, the network lifetime of AUV_PN and AEERP is less than those of CMDG schemes because the number of clusters is not increased proportionally with the network

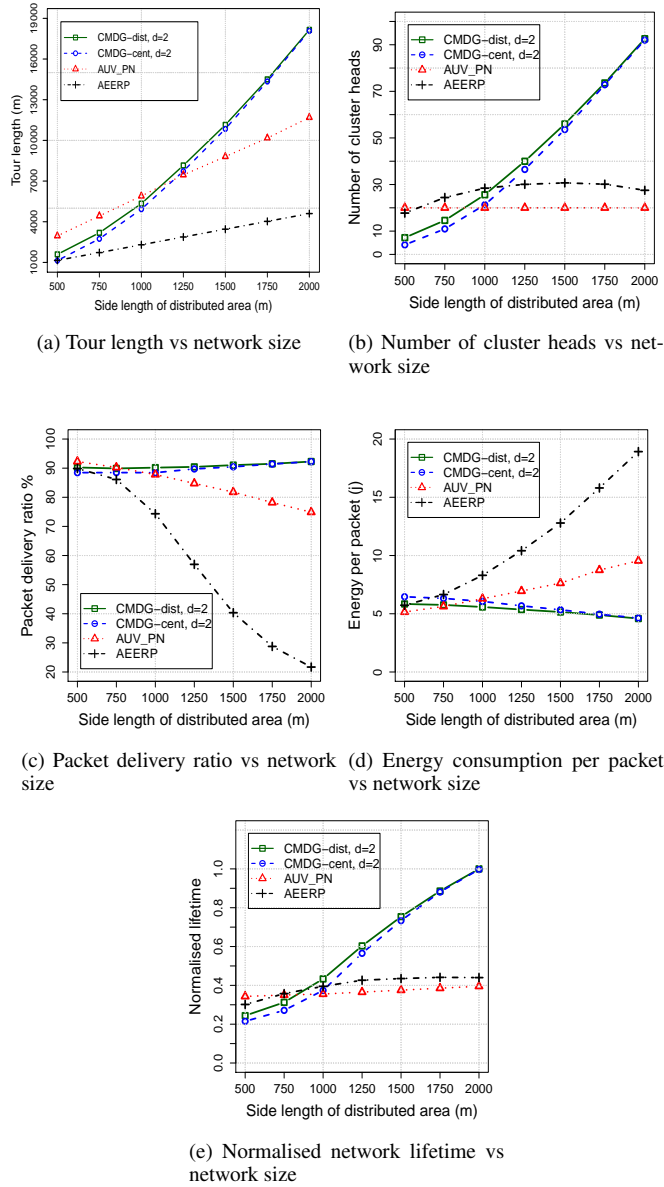


FIGURE 9: The impact of network area size on the mobile data gathering schemes

dimensions.

VII. CONCLUSION

AUV-aided mobile data gathering schemes have obtained considerable attention for environmental monitoring in UWSNs. In this paper, a new mobile data gathering scheme, CMDG, has been proposed to make a trade-off between data gathering latency and energy consumption. In a centralised or a distributed manner, CMDG selects a group of sensors as CHs by considering the energy and latency constraints. The cluster heads have the responsibility to collect the information from sensors and deliver them to the AUV when it arrives. After obtaining the list of CHs, the AUV plans a near-optimal tour to visit CHs and gather the data from

them and finally return and upload data to the static sink. If the mobility is presented to the system, CMDG still can keep its performance high for a long time. CMDG can improve the scalability and resolve the void problem in sparse networks. The simulation results indicated that CMDG can obtain a better trade-off between the data gathering latency and energy saving in comparison to those of existing mobile data gathering schemes. As future work, we plan to design mobile data gathering schemes with multiple AUVs and sinks in large-scale underwater sensor networks.

REFERENCES

- [1] J. Heidemann, W. Ye, J. Wills, A. Syed, and Y. Li, "Research challenges and applications for underwater sensor networking," in *Wireless Communications and Networking Conference*, Las Vegas, USA, pp. 228–235, IEEE, 2006.
- [2] E. Felemban, F. K. Shaikh, U. M. Qureshi, A. A. Sheikh, and S. B. Qaisar, "Underwater sensor network applications: A comprehensive survey," *International Journal of Distributed Sensor Networks*, vol. 2015, Article ID 896832, 14 pages, 2015. doi:10.1155/2015/896832.
- [3] M. R. Arshad, "Recent advancement in sensor technology for underwater applications," *Indian Journal of Marine Sciences*, vol. 38, no. 3, pp. 267–273, 2009.
- [4] S. M. Ghoreishi, A. Shahrabi, and T. Boutaleb, "Void-handling techniques for routing protocols in underwater sensor networks: Survey and challenges," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 2, pp. 800–827, 2017.
- [5] J. M. Jornet, M. Stojanovic, and M. Zorzi, "Focused beam routing protocol for underwater acoustic networks," in *Proceedings of the third ACM international workshop on Underwater Networks*, San Francisco, California, USA, pp. 75–82, ACM, 2008.
- [6] S. M. Ghoreishi, A. Shahrabi, and T. Boutaleb, "A novel cooperative opportunistic routing scheme for underwater sensor networks," *Sensors*, vol. 16, no. 3, p. 297, 2016.
- [7] D. Muhammed, M. H. Anisi, M. Zareei, C. Vargas-Rosales, and A. Khan, "Game theory-based cooperation for underwater acoustic sensor networks: Taxonomy, review, research challenges and directions," *Sensors*, vol. 18, no. 2, p. 425, 2018.
- [8] M. Ayaz, I. Baig, A. Abdullah, and I. Faye, "A survey on routing techniques in underwater wireless sensor networks," *Journal of Network and Computer Applications*, vol. 34, no. 6, pp. 1908–1927, 2011.
- [9] N. Javaid, S. Cheema, M. Akbar, N. Alrajeh, M. S. Alabed, and N. Guizani, "Balanced energy consumption based adaptive routing for iot enabling underwater wsns," *IEEE Access*, vol. 5, no. 1, pp. 40–51, 2017.
- [10] J. U. Khan and H.-S. Cho, "Data-gathering scheme using auvs in large-scale underwater sensor networks: A multihop approach," *Sensors*, vol. 16, no. 10, p. 1626, 2016.
- [11] S. Yoon and C. Qiao, "Cooperative search and survey using autonomous underwater vehicles (auvs)," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 3, pp. 364–379, 2011.
- [12] B. Zhang, Y. Wang, H. Wang, X. Guan, and Z. Zhuang, "Tracking a duty-cycled autonomous underwater vehicle by underwater wireless sensor networks," *IEEE Access*, vol. 5, pp. 18016–18032, 2017.
- [13] N. Javaid, N. Ilyas, A. Ahmad, N. Alrajeh, U. Qasim, Z. A. Khan, T. Liaqat, and M. I. Khan, "An efficient data-gathering routing protocol for underwater wireless sensor networks," *Sensors*, vol. 15, no. 11, pp. 29149–29181, 2015.
- [14] A. Khan, I. Ahmedy, M. H. Anisi, N. Javaid, I. Ali, N. Khan, M. Alsaqer, and H. Mahmood, "A localization-free interference and energy holes minimization routing for underwater wireless sensor networks," *Sensors*, vol. 18, no. 1, p. 165, 2018.
- [15] M. Akbar, N. Javaid, A. H. Khan, M. Imran, M. Shoaib, and A. Vasilakos, "Efficient data gathering in 3d linear underwater wireless sensor networks using sink mobility," *Sensors*, vol. 16, no. 3, p. 404, 2016.
- [16] M. H. Anisi and A. H. Abdullah, "Efficient data reporting in intelligent transportation systems," *Networks and Spatial Economics*, vol. 16, no. 2, pp. 623–642, 2016.
- [17] M. Zhao and Y. Yang, "Bounded relay hop mobile data gathering in wireless sensor networks," *IEEE Transactions on Computers*, vol. 61, no. 2, pp. 265–277, 2012.

- [18] D. Sandeep and V. Kumar, "Review on clustering, coverage and connectivity in underwater wireless sensor networks: a communication techniques perspective," *IEEE Access*, vol. 5, pp. 11176–11199, 2017.
- [19] M. Ma, Y. Yang, and M. Zhao, "Tour planning for mobile data-gathering mechanisms in wireless sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 4, pp. 1472–1483, 2013.
- [20] E. Saad, M. Awadalla, and R. Darwish, "A data gathering algorithm for a mobile sink in large-scale sensor networks," in *The Fourth International Conference on Wireless and Mobile Communications*, Sofia, Bulgaria, pp. 207–213, IEEE, 2008.
- [21] M. Ma and Y. Yang, "Sencar: an energy-efficient data gathering mechanism for large-scale multihop sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 10, 2007.
- [22] G. Abdul-Salaam, A. H. Abdullah, and M. H. Anisi, "Energy-efficient data reporting for navigation in position-free hybrid wireless sensor networks," *IEEE Sensors Journal*, vol. 17, no. 7, pp. 2289–2297, 2017.
- [23] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "An opportunistic void avoidance routing protocol for underwater sensor networks," in *30th International Conference on Advanced Information Networking and Applications*, Crans-Montana, Switzerland, pp. 316–323, IEEE, 2016.
- [24] P. Yao and S. Zhao, "Three-dimensional path planning for auv based on interfered fluid dynamical system under ocean current (june 2018)," *IEEE Access*, vol. 6, pp. 42904–42916, 2018.
- [25] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "An efficient auv-aided data collection in underwater sensor networks," in *32nd International Conference on Advanced Information Networking and Applications (AINA)*, Pedagogical University of Cracow, Poland, pp. 281–288, IEEE, 2018.
- [26] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "A cluster-based mobile data-gathering scheme for underwater sensor networks," in *2018 International Symposium on Networks, Computers and Communications (ISNCC)*, pp. 1–6, IEEE, 2018.
- [27] Y. Noh, U. Lee, P. Wang, B. S. C. Choi, and M. Gerla, "Vapr: void-aware pressure routing for underwater sensor networks," *IEEE Transactions on Mobile Computing*, vol. 12, no. 5, pp. 895–908, 2013.
- [28] U. Lee, P. Wang, Y. Noh, F. Vieira, M. Gerla, and J.-H. Cui, "Pressure routing for underwater sensor networks," in *29th conference on Information communications (INFOCOM)*, San Diego, California, USA, pp. 1676–1684, 2010.
- [29] M. T. Kheirabadi and M. M. Mohamad, "Greedy routing in underwater acoustic sensor networks: a survey," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 701834, 21 pages, 2013. doi:10.1155/2013/701834.
- [30] Z. Wang, G. Han, H. Qin, S. Zhang, and Y. Sui, "An energy-aware and void-avoidable routing protocol for underwater sensor networks," *IEEE Access*, vol. 6, pp. 7792–7801, 2018.
- [31] Y.-S. Chen and Y.-W. Lin, "Mobicast routing protocol for underwater sensor networks," *IEEE Sensors Journal*, vol. 13, no. 2, pp. 737–749, 2013.
- [32] A. Ahmad, A. Wahid, and D. Kim, "Aeerp: Auv aided energy efficient routing protocol for underwater acoustic sensor network," in *Proceedings of the 8th ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, pp. 53–60, ACM, 2013.
- [33] S. Yoon, A. K. Azad, H. Oh, and S. Kim, "Aurp: An auv-aided underwater routing protocol for underwater acoustic sensor networks," *Sensors*, vol. 12, no. 2, pp. 1827–1845, 2012.
- [34] J. U. Khan and H.-S. Cho, "A distributed data-gathering protocol using auv in underwater sensor networks," *Sensors*, vol. 15, no. 8, pp. 19331–19350, 2015.
- [35] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "An underwater routing protocol with void detection and bypassing capability," in *31st International Conference on Advanced Information Networking and Applications (AINA)*, Taipei, Taiwan, pp. 530–537, IEEE, 2017.
- [36] V. Chandrasekhar, W. K. Seah, Y. S. Choo, and H. V. Ee, "Localization in underwater sensor networks: survey and challenges," in *Proceedings of the 1st ACM international workshop on Underwater networks*, Los Angeles, USA, pp. 33–40, ACM, 2006.
- [37] M. B. Porter and H. P. Buckner, "Gaussian beam tracing for computing ocean acoustic fields," *The Journal of the Acoustical Society of America*, vol. 82, no. 4, pp. 1349–1359, 1987.
- [38] H. U. Yildiz, V. C. Gungor, and B. Tavli, "Packet size optimization for lifetime maximization in underwater acoustic sensor networks," *IEEE Transactions on Industrial Informatics*, 2018.
- [39] Y. Kim, Y. Noh, and K. Kim, "Rar: Real-time acoustic ranging in underwater sensor networks," *IEEE Communications Letters*, vol. 21, no. 11, pp. 2328–2331, 2017.
- [40] J. Llor and M. P. Malumbres, "Underwater wireless sensor networks: how do acoustic propagation models impact the performance of higher-level protocols?," *Sensors*, vol. 12, no. 2, pp. 1312–1335, 2012.
- [41] S. R. Thompson, Sound propagation considerations for a deep-ocean acoustic network. PhD thesis, Monterey, California. Naval Postgraduate School, 2009.
- [42] I. F. Akyildiz, D. Pompili, and T. Melodia, "Challenges for efficient communication in underwater acoustic sensor networks," *ACM Sigbed Review*, vol. 1, no. 2, pp. 3–8, 2004.
- [43] J. J. Bentley, "Fast algorithms for geometric traveling salesman problems," *ORSA Journal on computing*, vol. 4, no. 4, pp. 387–411, 1992.
- [44] M. Bellmore and G. L. Nemhauser, "The traveling salesman problem: a survey," *Operations Research*, vol. 16, no. 3, pp. 538–558, 1968.
- [45] C. Nilsson, "Heuristics for the traveling salesman problem," tech. rep., Tech. Report, Linköping University, Sweden, 2003.
- [46] A. Caruso, F. Paparella, L. F. M. Vieira, M. Erol, and M. Gerla, "The mandering current mobility model and its impact on underwater mobile sensor networks," in *The 27th Conference on Computer Communications, INFOCOM*, pp. 221–225, IEEE, 2008.
- [47] A. Akbas, H. U. Yildiz, B. Tavli, and S. Uludag, "Joint optimization of transmission power level and packet size for wsn lifetime maximization," *IEEE Sensors Journal*, vol. 16, no. 12, pp. 5084–5094, 2016.
- [48] H. U. Yildiz, "The impact of data fragmentation on network lifetime in underwater acoustic sensor networks," in *2018 26th Signal Processing and Communications Applications Conference (SIU)*, pp. 1–4, IEEE, 2018.
- [49] "Cplex." <https://www.ibm.com/products/ilog-cplex-optimization-studio>. 2018.
- [50] S. M. Ghoreyshi, A. Shahrabi, and T. Boutaleb, "A stateless opportunistic routing protocol for underwater sensor networks," *Wireless Communications and Mobile Computing*, vol. 2018, 2018.

...